Standard operating procedure (version 1.3) for read mapping, expression normalization, and data analysis in Functional Genomics core

1 Short read mapping (RNA-seq reads from Illumina)

1.1 Combine reference genomes and build `bowtie` index

First, a combined reference genome index is built. The combined reference genome includes the parasite, the host, and the spike-in control RNA. Both of these genomes are the most recent versions present in iRODS. The three *.fasta files are concatenated into one *.fasta file using the `cat` command.

`bowtie2-build` (version 2.1.0) takes the merged *.fasta file and makes a reference index. Default parameters are used. The output for this step is a `bowtie2` index which is a group of binary files used downstream in the read mapping step (with `tophat2`).

1.2 Build transcriptome index for `tophat`

The transcriptome annotation index is made by first converting the provided *.gff (gene annotation) files to *.gtf using `cuffcompare`, which is a part of the `cufflinks` package (version 2.1.1). The *.gff gene annotations files for both the host and the parasite are obtained from the iRODS server. A known issue during this conversion process is that genomic loci annotated as separate genes in the original *.gff files are merged when one gene is entirely contained within another gene or if two genes share a common exon.

`tophat2` (v2.1.0) is used to build the transcriptome index by supplying the `-G` and `--transcriptome-index=` options. The parameter passed to the `-G` option is the cuffmerge-produced *.gtf file. Default parameters are used. The output for this step is a `bowtie2` transcriptome index which is a group of binary files used downstream in the read mapping step.

1.3 Map the reads with `tophat2`

The raw fastq files from each library (one library per sample) are mapped to a combined reference genome index using `tophat2` (v2.0.9). Default options for `tophat2` are used with the following exceptions:

`-p 10` : this option is used to increase the speed of the mapping step by using multiple threading. It should have no effect on the resulting mapping locations.

`--library-type fr-secondstrand` : used because our reads are stranded using the Illumina protocol; this shouldn't have an effect on the resulting mapping locations, but it will be useful for downstream analysis.

`--transcriptome-index=` : this option improves the mapping accuracy especially across splice junctions.

example command:

```
tophat2 -p 10 -o /nv/hp10/klee370/scratch/malaria/Experiment13/pipeline.
    ↪ E13T07YSMmDpXXWB_GTCCGCL007R1_2010300.fastq.gz.secondstrand.pipeline --library-type
    ↪ fr-secondstrand --transcriptome-index=/nv/hp10/klee370/scratch/malaria/02
    ↪ _reference_genomes/combined_index/bowtie_transcriptome/known /nv/hp10/klee370/
    ↪ scratch/malaria/02_reference_genomes/combined_index/MmV2_Pcyno_spike-
```

```
↪ in_combined_index /nv/hp10/vnayak3/scratch/mahpic/fxngenomics/gibson/fromUGA/
↪ Experiment13/E13YerkesSequencing/E13YSTimePoint07/
↪ E13T07YSMmDpXXWB_GTCCGCL007R1_2010300.fastq.gz /nv/hp10/vnayak3/scratch/mahpic/
↪ fxngenomics/gibson/fromUGA/Experiment13/E13YerkesSequencing/E13YSTimePoint07/
↪ E13T07YSMmDpXXWB_GTCCGCL007R2_2010300.fastq.gz
```

The output for this step includes the accepted_hits.bam file, one line for each mapped read. Example output for three reads:

```
HWI-1KL135:72:C3GG6ACXX:8:1204:17547:23291 163 ERCC-00002 1 50
↪ 101M = 128 228
↪ TCCAGATTACTTCCATTTCCGCCCAAGCTGCTCACAGTATACGGGCGTCGGCATCCAGACCGTCGGCTGATCGTGGTTTTACTAGGCTAG
↪ @CCFFDADHHHHHJJIIJJJJJGIJJJJJJJJJEHIJJ>G<BGHJGJJIEHBEDEEE;CACDDBDDDDDDDDDBDDD?
↪ @DDDDDACCCDDDDDCACD@>BBB AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101
↪ YT:Z:UU NH:i:1 XS:A:-
HWI-1KL135:72:C3GG6ACXX:8:1201:20021:108943 163 ERCC-00002 2 50
↪ 101M = 49 148
↪ CCAGATTACTTCCATTTCCGCCCAAGCTGCTCACAGTATACGGGCGTCGGCATCCAGACCGTCGGCTGATCGTGGTTTTACTAGGCTAGA
↪ @CCFFFFFHFHHHGIIJHIGGJIJJGHIGIFIJIJICGIICGGIIBAG>EEBDEDEEDDDBD/;@>@B?@CB??>A?283:
↪ @C3?CCDDC3@C99>>?9@? AS:i:0 XN:i:0 XM:i:0 XO:i:0 XG:i:0 NM:i:0 MD:Z:101
↪ YT:Z:UU NH:i:1 XS:A:-
HWI-1KL135:72:C3GG6ACXX:8:1305:21012:98951 163 ERCC-00002 2 50
↪ 101M = 5 104
↪ CCAGATTACTTCCATTTCCGCCCAAGCTGCTCACAGTATACGGGCGTCGGCATCCAGACCGTCGGCTGATCGTGGTTTTACTAGGCTAGA
↪ @B@FFFFFHHHHHJIIIJJHIIJIIFFI>GIIJIGICFHIJIJIIGIIHD:??
↪ DCC@CDDBDDB@BDDACCDDDDDBDDDDCDC39@AACC>>??<B<<9B AS:i:0 XN:i:0 XM:i:0 XO:i:0
↪ XG:i:0 NM:i:0 MD:Z:101 YT:Z:UU NH:i:1 XS:A:-
```

2 Using Spike-in control RNA for quality control

An amount of spike-in RNA mix has been added to all of the libraries. This mix contains known concentrations of each of 90 different RNAs of various lengths and GC proportions. All of the spike-in results are included in the metadata file associated to each library.

2.1 Assessing the accuracy of RNA-seq in each library

In order to assure that the read counts from each of the RNA-seq libraries accurately reflects the level of RNA in the starting materials, we perform a correlation analysis between the read counts obtained through RNA-seq for each library and the known concentration of the spike-in RNA mix.

A custom script counts the number of reads that map to each of the spike-in control RNAs. This number is divided by the length of the RNA since longer RNAs will produce more reads in the libraries. Then the correlations between known spike-in concentration and experimental counts are produced and reported for each library. Both the known concentrations and the empirical expression levels are log-transformed since the RNA concentrations range across five orders of magnitude.

2.2 Preservation of strandedness in the library preparation

In the custom script mentioned above, a count is also maintained for the number of reads that map to the anti-sense strand of the spike-in RNA. The proportion of anti-sense and fusion transcripts for the spike-in RNAs are recorded for each library. This number will be very low (less than 1%) if strandedness was preserved in the library preparation protocol.

2.3 Number of reads that support fusion transcripts

In the custom script mentioned above, a count is also maintained for the number of reads that show evidence of a fusion transcript between a spike-in RNA and another contig, which could either be another spike-in RNA or a chromosome from one of the reference genomes. The proportion of reads supporting a fusion transcript for the spike-in RNAs are recorded for each library. This number should be very low (less than 1%).

3 5'-to-3' coverage uniformity

For various reasons, RNA-seq read depth coverage can decrease when moving from the 3' end to the 5' end of transcripts. Therefore, another important quality control step is to make sure that coverage across the length of transcripts does not drop off precipitously. To accomplish this task, we use `RSeqC` version 2.3.8, a quality control suite for next-generation sequencing.

The inputs to RSeqC are the bam file from `tophat2` and a bed file containing the locations of the spike-in control RNAs. The transcripts used to determine this coverage evenness are only the spike-in controls. This is because the reference gene annotations from *Macaca mulatta* are incomplete, and mis-annotations can lead to inaccurate signals in the coverage plots. Default parameters are used.

The output from the `RSeqC` step includes a text file from which the number of reads mapping to each percentile across all transcripts is present.

```
percentile      count
0       3003
1       22622
2       45348
3       64643
4       85990
5       98411
6       110037
...
47      224050
```

```
48      222860
49      224618
50      221901
51      227525
52      228956
53      226905
...
94      24891
95      20850
96      13360
97      7689
98      3408
99      187
100     0
```

Using the output from `RSeqC`, the average number of reads mapping to the first 50% of the reads (the 5' end) is then divided by the average number of reads mapping to the second 50% of the reads (the 3' end); subsequently, the value is log-base-2-transformed. This value is the 5':3' coverage evenness ratio.

4 Gene expression quantification using `htseq`

For each library, the gene expression levels are calculated independently.

Sort the output bam files (from `tophat2`, accepted_hits.bam) by read name; this is the required input format for `htseq`.

Run `htseq` (version 0.5.4) to quantify the number of reads mapping to each gene in the reference annotation file, which was made in the previous step. The two required parameters are the bam file (sorted by name) and the *.gtf (gene annotation) file.

default parameters are used with the following exception: `-s reverse` : this parameter is used since the Illumina protocol to preserved strandedness uses the second (reverse) strand. The mapping input is the sam file sorted by read name (and not by chromosomal location). The reference annotation is the same as that from the read mapping step.

example command:

```
samtools view /nv/hp10/klee370/scratch/malaria/Experiment13/pipeline.
    ↪ E13T07YSMmDpXXWB_GTCCGCL007R1_2010300.fastq.gz.secondstrand.pipeline/accepted_hits.
    ↪ bam_sorted.bam | /nv/hp10/klee370/data/data/bin/HTSeq-0.5.4p5/scripts/htseq-count -s
    ↪  reverse - /nv/hp10/klee370/scratch/malaria/02_reference_genomes/
    ↪ Macaca_mulatta_rheMac2/version_2_0/cuffcmp.combined.gtf > /nv/hp10/klee370/scratch/
    ↪ malaria/Experiment13/pipeline.E13T07YSMmDpXXWB_GTCCGCL007R1_2010300.fastq.gz.
    ↪ secondstrand.pipeline/accepted_hits.bam_htseq_counts
```

The output of the command is a flat file (one for each library):

XLOC_000001 250
XLOC_000002 30
XLOC_000003 215
XLOC_000004 445
XLOC_000005 128
XLOC_000006 32
...
XLOC_015441 222
XLOC_015442 19
XLOC_015443 194
XLOC_015444 520

Each 'XLOC_' gene locus corresponds to a gene annotated in the *.gtf file. After all libraries have had their expression quantified, the individual flat files are merged into one large flat file.

5 Exon expression quantification using `dexseq_count.py`

Before exon-level quantification, a flattened *.gff file must be made using `dexseq_prepare_annotation.py`. Note: when the `dexseq_prepare_annotation.py` script is used to flatten the annotation file, some of the resulting exons will not be identical in length and genomic coordinates to the original annotations.

Output bam files from tophat (accepted_hits.bam) are sorted by read name; (this is the required imput format for `dexseq_count.py`).

`dexseq_count.py` is used to quantify the number of reads mapping to each exon in the flattened reference annotation file.

default parameters are used with the following exception: `-s reverse` : this parameter is used since the Illumina protocol to preserved strandedness uses the second (reverse) strand. The mapping input is the sam file sorted by read name (and not by chromosomal location). The reference annotation is the same as that from the read mapping step.

example command (*.gff flattening):

```
python /nv/hp10/klee370/R/x86_64-unknown-linux-gnu-library/2.15/DEXSeq/python_scripts/
    ↪ dexseq_prepare_annotation.py /nv/hp10/klee370/scratch/malaria/02_reference_genomes/
    ↪ Macaca_mulatta_rheMac2/version_2_0/cuffcmp.combined.gtf /nv/hp10/klee370/scratch/
    ↪ malaria/02_reference_genomes/Macaca_mulatta_rheMac2/version_2_0/cuffcmp.combined.
    ↪ HTSeq_flattened.gff
```

example command (exon-level expression quantification):

```
samtools view /nv/hp10/klee370/scratch/malaria/Experiment13/pipeline.
    ↪ E13T02YSMmDpXXWB_CGATGTL006R1_2010267.fastq.gz.secondstrand.pipeline/accepted_hits.
    ↪ bam_sorted.bam | python /nv/hp10/klee370/R/x86_64-unknown-linux-gnu-library/2.15/
    ↪ DEXSeq/python_scripts/dexseq_count.py -p yes -s reverse /nv/hp10/klee370/scratch/
    ↪ malaria/02_reference_genomes/Macaca_mulatta_rheMac2/version_2_0/cuffcmp.combined.
```

```
↪ HTSeq_flattened.gff - /nv/hp10/klee370/scratch/malaria/Experiment13/pipeline.
↪ E13T02YSMmDpXXWB_CGATGTL006R1_2010267.fastq.gz.secondstrand.pipeline/accepted_hits.
↪ bam_htseq_EXON_counts_reverse
```

The output of the command is a flat file (one for each library):

```
XLOC_000001:001  0
XLOC_000001:002  19
XLOC_000001:003  33
XLOC_000001:004  41
XLOC_000001:005  19
XLOC_000001:006  25
...
XLOC_015444:004  65
XLOC_015444:005  97
XLOC_015444:006  55
XLOC_015444:007  56
XLOC_015444:008  35
```

Each line represents an exon from a gene, the name of which corresponds to a gene annotated in the *.gff file. After all libraries have had their expression quantified, the individual flat files are merged into one large flat file.

6 Library size correction (normalization)

After quantifying the gene expression levels, the next step in the analysis is identification of genes that are differentially expressed across the experimental conditions.

Since some libraries may have been sequenced more deeply than others, the libraries need to be made comparable (normalized) before determining differential gene expression between libraries. Gene expression normalization and exon expression normalization are performed using the gene and exon level expression files with `DESeq` and `DEXSeq`, respectively. `DESeq` version 1.10.1; `DEXSeq` version 1.4.0. Both are available in the `bioconductor` suite in `R`. Default parameters are used.

7 Reporting of the mapping data

Within four weeks of receiving all of the short read data for a given experiment, the mapped reads files (*.bam) as well as the expression quantification (for raw counts, library-size normalized expression levels, and SNM normalized expression (for both removing animal effect as well as simply adjusting for animal effect)) levels for both gene-level and exon-level mappings, and the meta-data file from the experiment, which includes the quality control results described above, will be transferred to UGA Informatics core by copying the data files into the pipeline area on biocluster per the Informatics Data Transfer SOP. Differential gene expression and gene set enrichment analysis results will be reported at a later time.

8 Differential gene expression (DGE)

After library size correction, the expression levels are log-2-transformed and imported into JMP Genomics (version 6.0) which is a GUI that uses SAS (version 9.3) for its statistical back-end. DGE is assessed using an ANOVA statistical framework in JMP. For between-animal differences, a one-way ANOVA using "animal" as the factor is used. For between-TP differences, an ANOVA is performed using "animal" as a random effect and "time" or "drug" as the fixed effect. A false discovery rate of 5% is used to identify differentially expressed genes.

9 SNM normalization

After log2 transformation, we used the Bioconductor package in R for SNM (supervised normalization of microarrays), to normalize our expression data. Each set of expression-level data is SNM normalized in two ways: 1) animal effect is adjusted, 2) animal effect is removed. In both situations, the adjustment variable is the animal and the biological variable of interest is the timepoint.

10 Gene set enrichment analysis

After performing differential gene expression analysis, the Gene Set Enrichment Analysis (GSEA) suite (version 2.0) from the Broad Institute is used to identify gene sets (KEGG pathways, Gene Ontology terms, transcription factor binding site motifs, etc.) that are enriched in one condition compared to another. Ranked gene lists are used as input for GSEA's pre-ranked gene list module; genes are ranked using the t-statistics for contrasts of interest. Instead of only considering the genes previously identified as differentially expressed GSEA's pre-ranked list module uses all of the genes in the annotation. Positively and negatively enriched pathways are reported. The GSEA software default cut-off of FDR=25% will be used.

The results from this step (i.e. significantly altered gene sets) are reported as a list of gene sets, their respective q-values and enrichment scores.